Integrating a Non-Uniformly Sampled Software Retina with a Deep CNN Model

Piotr Ozimek
piotrozimek9@gmail.com
J. Paul Siebert
http://www.dcs.gla.ac.uk/~psiebert

School of Computing Science
Sir Alwyn Williams Building
University of Glasgow
Glasgow G12 8RZ
SCOTLAND
United Kingdom

Abstract

We present a biologically inspired method for pre-processing images applied to CNNs that reduces their memory requirements while increasing their invariance to scale and rotation changes. Our method is based on the mammalian retino-cortical transform: a mapping between a pseudo-randomly tessellated retina model (used to sample an input image) and a CNN. The aim of this first pilot study is to demonstrate a functional retina-integrated CNN implementation and this produced the following results: a network using the full retino-cortical transform yielded an F1 score of 0.80 on a test set during a 4-way classification task, while an identical network not using the proposed method yielded an F1 score of 0.86 on the same task. The method reduced the visual data by ~×7, the input data to the CNN by 40% and the number of CNN training epochs by 64%. These results demonstrate the viability of our method and hint at the potential of exploiting functional traits of natural vision systems in CNNs.

1 Introduction

We present a first study into improving the efficiency of image analysis using CNNs by pre-processing using a software-based retina model, similar in structure to those of mammals and humans, to substantially reduce the visual input data size to the CNN and also simplify its learning requirements. Our retina model samples the input image using Gaussian receptive fields located on a space-variant (foveated) pseudo-random sampling tessellation generated by annealing. This data is then spatially transformed using a polar transform to generate a *cortical map*, similar to that observed in the human visual system. This mapping splits the visual field into two hemifields, as observed in the brain, and these are projected into a regular image suitable for processing by a Keras CNN model we formulated. This mapping not only reduces the visual data by a factor of ~×7, it also affords a degree of input image scale and rotation invariance and therefore has the potential to both reduce the network size substantially and also simplify its learning requirements.

The key challenge of this study, proposed in [20], is how to transform the irregularly distributed retina samples to a matrix format that can be processed within conventional CNN environments and then to devise a CNN architecture which is compatible with this input.

We demonstrate that our software retina-integrated CNN formulation is capable of learning object classes, affords a reduction in network size and also requires fewer training epochs to achieve a classification performance similar to that of a standard CNN formulation.

2 Background

2.1 The Mammalian Vision System

Any perceived light entering the eye-ball stimulates a hemispherical layer of photoreceptor cells. These cells are densely packed in the central region of the retina (*the fovea*) and are more sparsely distributed in its peripheries [5].

The signals produced by photoreceptor cells are sequentially pre-processed through up to 4 different neuron types before leaving the retina and reaching the brain. It is worth noting that the topologies of these 'intermediate' retinal neurons coarsely follow the foveated topology of photoreceptor cells, and that it is this topological organisation combined with a visual attention mechanism that enables the retina to vastly cull the redundant information passed onto the brain [4].

The final retinal neurons that relay the visual signal to the brain (*V1*) via retinal *ganglion cells* (*RGCs*). Most RGCs are individually connected to local clusters of multiple neighbouring photoreceptor cells to form what is termed the RGC's *receptive field*. The sizes of these receptive fields increase with eccentricity, with the foveal RGCs relaying information from individual photoreceptor cells [11]. Individual RGCs have different receptive field response profiles depending on their function, which can range from discerning detail to computing the magnitude of differential motion [15] [16]. The signal from each eye-ball is split into two halves which are projected separately onto V1, where they are translated to a complex logarithmic mapping similar to the one in Figure 1. A form of this mapping has been proposed as the basis on which our brains process vision and it could potentially contribute towards scale invariance in biological vision systems [18].

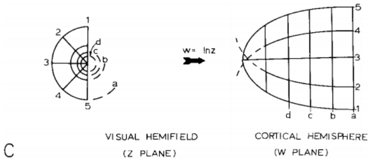


Figure 1: Global retinotopic mapping, taken from [18]

2.2 Computational Retina Models

A detailed expositon of the many computational models reported in the literature, e.g [2, 9, 12, 18], is beyond the scope of this paper. However, one recent model of note is that of Gobron et. al. [7] who model the coarse functional properties of the retina using a cellular automaton and GPU accelerate their model. Only the general function of the 5 different retinal neuron types has been expressed - architectural features of the retina such as the foveated topography of its neurons and their receptive field response profiles were not represented in this model and hence it is unsuitable for the purpose of our work.

Also of note, Pamplona et al. [17] have devised a method of generating foveated images using overlapping Gaussian receptive fields and have provided a way of performing conventional image processing functions on such images using matrix operations. However, their work appears to produce a number of visual artefacts in its implementation.

2.2.1 Balasuriya's Retina

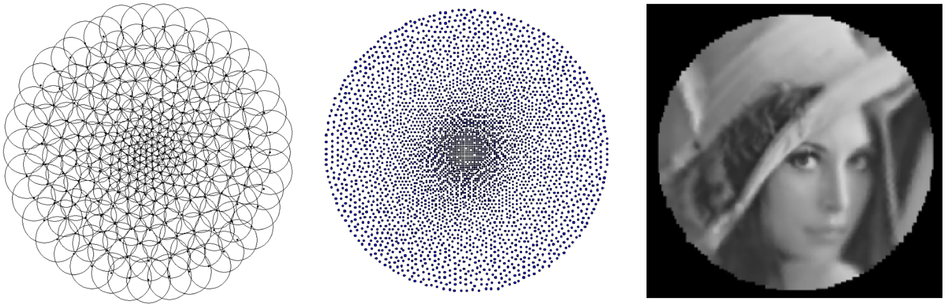


Figure 2: **Left:** Gaussian receptive fields on top of a retina tessellation, taken from [1]. **Centre:** The 4196 node tessellation used in this paper. **Right:** A backprojected retinal image.

The retina model that has been employed in this paper was developed by Balasuriya [1] whose work investigates the generation, sampling function, feature extraction and gaze control mechanism of a self-organized software retina.

To generate the retina tessellation without local discontinuities, distortions or other artefacts Balasuriya employs a self-similar neural network as described by Clippingdale & Wilson [3]. This method relies on a network of N nodes jointly undergoing random translations to produce a tessellation with a near-uniform dense foveal region that seamlessly transitions into a sparse periphery. Each node in the resultant tessellation defines the location of a receptive field's centre. The receptive fields somewhat follow the biological retina's architecture; they all have a Gaussian response profile the standard deviation of which scales linearly as a function of local node density, which in turn scales with eccentricity. This scaling balances between introducing aliasing at the sparsely sampled peripheries and super-Nyquist sampling at the densely sampled foveal region.

The values sampled by the receptive fields are then stored in an *imagevector*, which is a one-dimensional array of intensity values which supply the remainder of his visual processing chain and are also used to feed the processing pipeline in this work.

3 The Retino-Cortical Transform

3.1 Retinal Sampling

Based on Balasuriya's reported parameterisations [1], we generated a retina tessellation, as described in Section 2.2.1, with $N = 4,196$ nodes and $r_{fov} = 0.2$ (the fovea's radius as a fraction of the tessellation's radius) employing $N_{iter} = 20,000$ annealing iterations for self-organisation of the retina sampling tessellation [3]. Unfortunately no guidelines have been provided by Balasuriya regarding the optimisation of the parameters that define the Gaussian receptive fields: a $dist_5$ variable defines the mean pixel distance of the 5 central foveal nodes to their 5 closest neighbours; a σ_{base} variable defines the base size and standard deviation of the Gaussian receptive fields; and finally, a σ_{ratio} defines the eccentricity scaling factor of the Gaussian receptive fields' standard deviation.

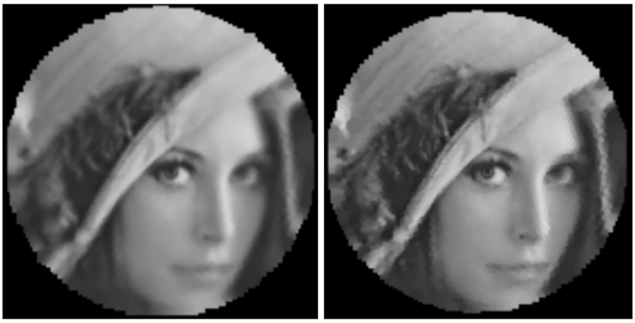


Figure 3: Backprojected retinal images. **Left:** a well-parametrized retina. **Right:** a badly parametrized retina. Note the jaggy edges at the peripheries of the right image.

A useful visualisation of the information captured by the retina is the *backprojected image* (Figure 2, right & Figure 3). It allows one to check whether the retinal subsampling is sufficiently sharp and free from aliasing artefacts. In order to obtain the backprojected image Gaussian receptive fields are projected onto an image-plane and scaled by their corresponding imagevector values. This image is then normalized by a Gaussian 'heatmap' image, which is a projection of the receptive field Gaussians onto an image plane, without any scaling, that reveals the density and uniformity of these sampling fields.

The receptive field parameters used were chosen manually by visually examining the Gaussian heatmaps and backprojected images for various parameter combinations. Attempts have been made at automating the process of optimising these parameters by trying to minimise various difference metrics between the backprojected image and the original image, however the process always favoured overly sharp retinas that produced excessive aliasing artefacts. Prioritising the foveal region in the optimisation process has also been unsuccessful, since the optimisation algorithm had no means of detecting aliasing artefacts outside the fovea. The (manually) chosen receptive field parameters are: $dist_5 = 1.0$, $\sigma_{base} = 0.4$ and $\sigma_{ratio} = 0.26$. The resultant retina's size is 168×168 px.

3.2 Cortical Image Generation

3.2.1 Requirements and Approach

The core idea behind cortical images is to first map the receptive field centres onto a new space and then project the associated imagevector intensities via Gaussian kernels centred on these locations, i.e. perform a *forward warp*. The approach taken eliminates the possibility of holes in the mapping as the size of the Gaussian projections can be increased to compensate.

The cortical images should ideally be *conformal*, i.e. preserve local angles and maintain a fairly uniform receptive field density while preserving local information captured by the retina without introducing any artefacts. These criteria must be satisfied to enable the convolution kernels of CNNs to extract features from the resultant cortical image. The literature reports sampling points at an adjusted log-polar space are the most appropriate retinocortical mapping, as it is believed they are employed in the primate visual cortex [18]. It is mathematically a plausible mapping for foveated images as it stretches out the fovea and squashes the peripheral field; it is also a conformal mapping.

3.2.2 The Cortical Mapping

Retinal log-polar coordinates consist of θ, which is the angle about the origin (the centre-most point of the fovea), and ρ, which is the log of the distance from the origin. The x and y variables below are Cartesian coordinates relative to the origin.

$$\rho = log\sqrt{x^2+y^2}\ ,\ \theta = atan2(y/x) \tag{1}$$

As evident in the left side of Figure 4 the log-polar space suffers from severe sparsity in the foveal region and excessive density at the peripheries. This has been mitigated by deviating from the approach proposed in the literature, removing the log operator from equation (1) and switching to the 'linear' polar space:

$$r = \sqrt{x^2+y^2} \tag{2}$$

The right side of Figure 4 demonstrates the drastic improvement in node uniformity by switching to the polar space, although the foveal region is still undesirably sparse and the extreme peripheries are packed in tight rows. The uniformity of the polar mapping also suffers at $r = 30$ where the node density is too high compared to other regions. These issues have been resolved by adopting the approach from the work of Schwartz [19] and adjusting the mapping with an α parameter while also splitting the retina tessellation vertically into two halves and mapping each half separately. This solves the singularity issue at the fovea and brings the mapping closer to the experimental data of activations in the visual cortices of different primates. The resultant coordinate equations for the cortical mappings are:

$$\mathbf{Y_{cort}} = \sqrt{(x+\alpha)^2+y^2}\ ,\ \mathbf{X_{cort}} = atan2(y/(x+\alpha)) \tag{3}$$

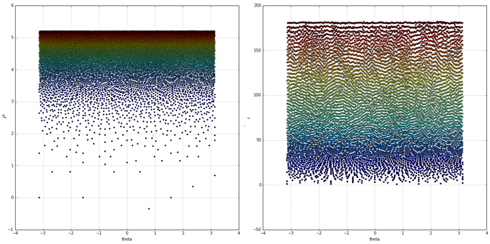


Figure 4: Colour coded receptive field centres mapped onto the log-polar (left) and linear-polar (right) spaces. Warmer colours indicate receptive fields closer to the peripheries, whereas colder colours indicate points closer to the fovea.

In Figure 5 the α parameter introduces a normalising transformation of the polar space. It is added to the x coordinate to virtually shift the tessellation's nodes away from the origin horizontally. In the polar space this manifests itself as all of the nodes being brought closer to $X = 0$, with the effect increasing logarithmically towards the foveal nodes at $Y = 0$. As the α parameter increases the peripheral nodes (red and dark blue in Figure 5) protrude

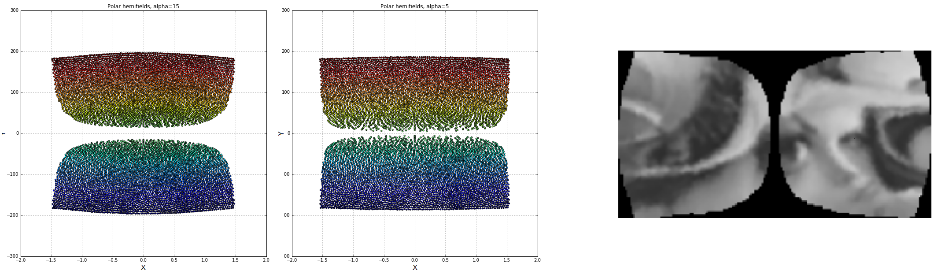


Figure 5: **Left:** Two hemifields of receptive field centres mapped onto a polar space with varying α values (15, 5). Colour-coded based on the value of $sign(x) * r$. **Right:** A cortical 'Lena' image equivalent to the retinal backprojection from the right side of Figure 2.

proportionately; this is desirable as it addresses the issue of tightly packed rows of nodes from Figure 4. Note that in order for the left half of the retina to mirror the right one in Figure 5 its coordinates have been adjusted as follows:

$$\mathbf{X_{left}} = -\sqrt{(x-\alpha)^2+y^2}\ ,\ \mathbf{Y_{left}} = atan2(y/x-\alpha) - sign(atan2(y/x-\alpha)) * \pi \qquad (4)$$

It was decided that a value of $\alpha = 10$ will be used, as upon visual inspection it appeared to be the most uniform. Lower α values lead to an overly sparse foveal region, while higher values produced an overly dense region at $Y \approx \pm 70$, $X \approx 0$. In order to define the aspect ratio of cortical images the mean node distances along the x and y axes were equated.

Cortical images were produced by projecting Gaussian kernels scaled (in height) by the associated imagevector value onto the appropriate nodes' locations with a sub-pixel accuracy of 1 decimal place. The resultant image was then normalized by the cortical Gaussian heatmap image, much like when generating retinal backprojected images in Section 3.1. The cortical Gaussians were parameterised with $\sigma = 1.2$ and clipped at 7 pixels width.

The resultant cortical images, an example of which can be seen in Figure 5, satisfy all the criteria for an acceptable input to a CNN: local angles are preserved, receptive fields are projected at a sufficiently uniform density and most of the local information captured by the retina is preserved without introducing any noise or artefacts. The cortical images have a resolution of 179×96px, while a square that best fits the retina's resolution is 168×168px large. Accordngly, the retino-cortical mapping reduces the data input to the CNN by ~ 40%.

4 Validation

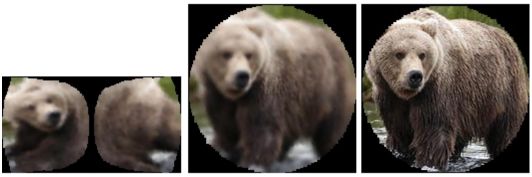
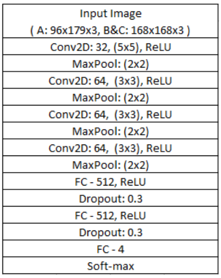


Figure 6: An example *Brown Bear* image from each of the three subsets (A, B and C).

A dataset suitable for training retina-integrated CNNs (RI-CNNs) was created by selecting and pre-processing the appropriate classes from ImageNet [6]. To prevent the classification task from being trivial, each object class in the dataset should share a subset of its visual features with at least one other class, meaning that the classes should be somewhat similar to each other.

In order to separately evaluate each part of the proposed retino-cortical transform three validation subsets have been constructed: subset A is made up of cortical images (Fig. 6, left), subset B is retinal backprojected images (Fig. 6, centre) and subset C consists of the conventional images, masked with the retinal lens (Fig. 6, right).

	Training	*Eval.*	*Test*	**TOTAL**
Hoop	2560	727	372	**3659**
Brown Bear	2422	693	350	**3465**
Keybrd.	2490	711	360	**3561**
Racoon	2492	704	339	**3535**
TOTAL	**9964**	**2835**	**1421**	**14220**



Input Image
(A: 96x179x3, B&C: 168x168x3)
Conv2D: 32, (5x5), ReLU
MaxPool: (2x2)
Conv2D: 64, (3x3), ReLU
MaxPool: (2x2)
Conv2D: 64, (3x3), ReLU
MaxPool: (2x2)
Conv2D: 64, (3x3), ReLU
MaxPool: (2x2)
FC - 512, ReLU
Dropout: 0.3
FC - 512, ReLU
Dropout: 0.3
FC - 4
Soft-max

Figure 7: **Left:** Per class and per split fixation image counts. The numbers are consistent across all 3 subsets of the dataset. **Right:** The CNN architecture used in this paper.

The object categories selected for the classification task are *Basketball Hoop*, *Brown Bear*, *Keyboard* and *Racoon*. The similarities between *Brown Bear* and *Racoon* (furry animal), *Basketball Hoop* and *Keyboard* (synthetic object with a grid-like key feature) helped ensure that the classification task is not trivial. The class objects were cropped out from their original images using the bounding boxes provided in ImageNet [6]. The resultant images passed automatic selection that ensured the images were not too small ($width, height > 75, 75$) or too long ($1/3 < width/height < 3$), and were then processed by appropriate parts of the retina pipeline to produce the three subsets.

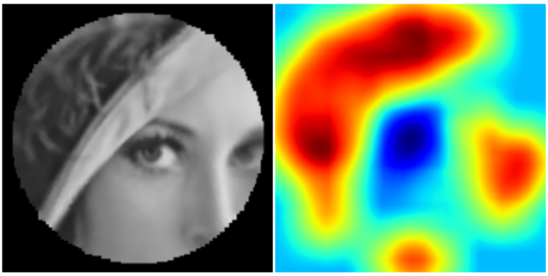


Figure 8: **Left:** A retinal backprojection image. **Right:** the equivalent saliency map. Note the inhibition near the foveal region.

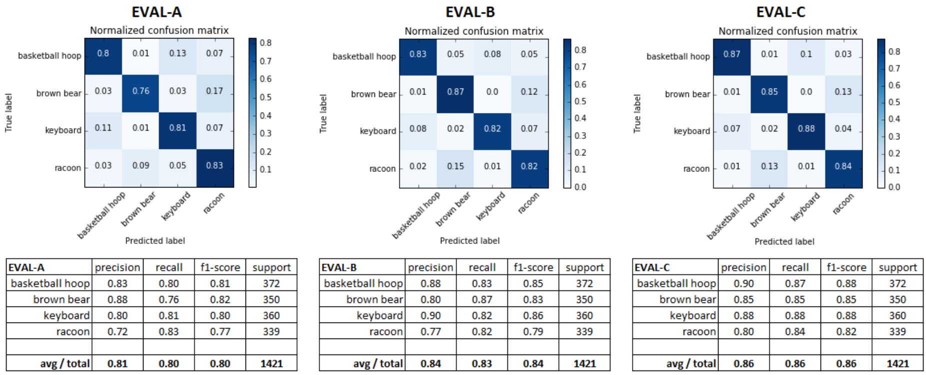
The image locations fixated on by the retina were selected by a gaze control system that is both simple and sufficient to meet the needs of this study. The algorithm maintained a saliency map driven by SIFT features that was inhibited at locations of past fixations (Figure 8, right). In order to correct a large imbalance between the class frequencies the number of retina fixations was varied per class. The final image counts in the dataset can be seen in Figure 7.

5 Results and Discussion

In order to evaluate in isolation the performance contributions of the retinal subsampling mechanism and the cortical image representation to the overall pipeline, three CNNs were trained using Keras 2.0.2, each with the same architecture but each using a different subset of the dataset built in the previous section. The CNN architecture used (Figure 7) was chosen by trialing various architectures to maximise their performance over the cortical image dataset. A relatively simple architecture was chosen in accordance the objectives of this study.

Adam [14] was employed for training optimisation in combination with categorical cross-entropy as the loss function. Early stopping callbacks (used to monitor improvements in validation accuracy) were employed to prevent unproductive training. L2 regularisation of strength $\lambda = 0.02$ was applied to the internal fully connected layers to prevent overfitting, however that value could have been increased as the model still displayed signs of overfitting. The key figures from training are:

- **Network EVAL-A**, using (96x179) cortical images, reached its peak performance **(validation loss= 0.605, validation accuracy=82.26%)** after **16 epochs**.
- **Network EVAL-B**, using (168x168) retinal backprojected images, reached its peak performance **(validation loss= 0.493, validation accuracy=86.14%)** after **21 epochs**.
- **Network EVAL-C**, using (168x168) conventional images, reached its peak performance **(validation loss= 0.488, validation accuracy=87.51%)** after **25 epochs**.



EVAL-A	basketball hoop	brown bear	keyboard	racoon
basketball hoop	0.8	0.01	0.13	0.07
brown bear	0.03	0.76	0.03	0.17
keyboard	0.11	0.01	0.81	0.07
racoon	0.03	0.09	0.05	0.83

EVAL-B	basketball hoop	brown bear	keyboard	racoon
basketball hoop	0.83	0.05	0.08	0.05
brown bear	0.01	0.87	0.0	0.12
keyboard	0.08	0.02	0.82	0.07
racoon	0.02	0.15	0.01	0.82

EVAL-C	basketball hoop	brown bear	keyboard	racoon
basketball hoop	0.87	0.01	0.1	0.03
brown bear	0.01	0.85	0.0	0.13
keyboard	0.07	0.02	0.88	0.04
racoon	0.01	0.13	0.01	0.84

EVAL-A	precision	recall	f1-score	support
basketball hoop	0.83	0.80	0.81	372
brown bear	0.88	0.76	0.82	350
keyboard	0.80	0.81	0.80	360
racoon	0.72	0.83	0.77	339
avg / total	**0.81**	**0.80**	**0.80**	**1421**

EVAL-B	precision	recall	f1-score	support
basketball hoop	0.88	0.83	0.85	372
brown bear	0.80	0.87	0.83	350
keyboard	0.90	0.82	0.86	360
racoon	0.77	0.82	0.79	339
avg / total	**0.84**	**0.83**	**0.84**	**1421**

EVAL-C	precision	recall	f1-score	support
basketball hoop	0.90	0.87	0.88	372
brown bear	0.85	0.85	0.85	350
keyboard	0.88	0.88	0.88	360
racoon	0.80	0.84	0.82	339
avg / total	**0.86**	**0.86**	**0.86**	**1421**

Figure 9: Confusion matrices and different performance metrics of the three CNNs evaluated against the appropriate test sets.

The results from evaluating the networks against the test set (Figure 9) show that both applying the full retino-cortical transform and the retinal subsampling lead to a small decrease in the CNNs' performance. The network trained on conventional images performed the best, with an average F1 score of 0.86; the network trained on retinal images landed an F1 score of 0.84 while the cortical images network had an F1 score of 0.80 showing that remapping the image from the retinal to the cortical space was the most damaging aspect of the retino-cortical transform. As seen in the matrices in Figure 9, the majority of the networks' confusion is between the classes sharing similar key features. Although the retina has reduced classification performance, the gap between the performance scores of the different networks is modest, the required training epochs have been reduced significantly and the network EVAL-A has successfully demonstrated the learning capacity of convolutional neural networks for images in the cortical view.

6 Conclusions & Further Work

This work has presented a pilot study into a novel method for pre-processing images provided to CNNs. The method draws inspiration from the mammalian visual system by imitating the retino-cortical transform to reduce the networks' memory requirements as well as affording a degree of scale and rotation invariance. The contributions of our work comprise: a specific retina model, a coarsely optimised cortical transform formulation and an image classification dataset comprising three subsets designed to probe the impact of the spatial sampling and transformation components of the pipeline. Evaluating a CNN architecture on the three data subsets has shown that the performance of the retina-integrated CNN is comparable to that of a CNN working with conventional images, while image data reduction, network size reduction and learning simplification has been confirmed. To the best of the authors' knowledge no prior attempts have been made at integrating a similar process to CNNs.

This paper has laid the groundwork for further investigations into integrating the retino-cortical transform with convolutional neural networks. The authors propose to develop a custom non-shared CNN layer which is fed directly by the retina image vector, appropriately transformed into a 2D polar retinotopic mapping. We then propose to investigate more elaborate CNN architectures which are best suited for retino-cortical transformed input. Locally connected convolution layers, as well as streams of parallel convolutions each processing a separate portion of the cortical image, are both relevant features of CNN architectures that appear to be worth investigating.

Our current investigations include modelling the full gamut of known retinal ganglion cells, and the wider range of low level computations that are essential to high-level visual reasoning tasks related to edge detection, motion and prediction [8]. We are also considering more sophisticated gaze control algorithms, potentially based on learning and generating target specific saliency maps, as in Hong et a. [10].

Finally, we are working towards more extensive training datasets based on both static and video imagery captured using a camera mounted on a robot arm to support scene exploration and hand-eye visual serving. The Large Scale Video Classification Network [13]; as it employs a primitive form of foveation, therefore we are investigating the associated LSVC image datasets using our retina approach.

References

[1] Sumitha Balasuriya. *A Computational Model of Space-Variant Vision Based on a Self-Organized Artifical Retina Tesselation*. PhD thesis, Department of Computing Science, University of Glasgow, March 2006.

[2] Marc Bolduc and Martin D. Levine. A Real-Time Foveated Sensor with Overlapping Receptive Fields. *Real-Time Imaging*, 3(3):195–212, 1997. URL http://dx.doi.org/10.1006/rtim.1996.0056.

[3] Simon Clippingdale and Roland Wilson. Self-similar neural networks based on a kohonen learning rule. *Neural Networks*, 9(5):747–763, 1996.

[4] Christine A Curcio and Kimberly A Allen. Topography of ganglion cells in human retina. *Journal of comparative Neurology*, 300(1):5–25, 1990.

[5] Christine A Curcio, Kenneth R Sloan, Robert E Kalina, and Anita E Hendrickson. Human photoreceptor topography. *Journal of comparative neurology*, 292(4):497–523, 1990.

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 248–255. IEEE, 2009.

[7] Stephane Gobron, Francois Devillard, and Bernard Heit. Retina simulation using cellular automata and gpu programming. *Machine Vision and Applications*, 18(6):331–342, 2007.

[8] Tim Gollisch and Markus Meister. Eye smarter than scientists believed: neural computations in circuits of the retina. *Neuron*, 65(2):150–164, 2010.

[9] H. Gomes. *Model Learning in Iconic Vision*. PhD thesis, University of Edinburgh, School of Informatics, Edinburgh, Scotland, UK, 2002.

[10] Seunghoon Hong, Tackgeun You, Suha Kwak, and Bohyung Han. Online tracking by learning discriminative saliency map with convolutional neural network. In *ICML*, pages 597–606, 2015.

[11] David H Hubel, Janice Wensveen, and Bruce Wick. *Eye, brain, and vision*. Scientific American Library New York, 1995.

[12] Alan Johnston. The geometry of the topographic map in striate cortex. *Vision Research*, 29(11):1493–1500, 1989. ISSN 0042-6989. doi: 10.1016/0042-6989(89) 90133-8. URL http://www.sciencedirect.com/science/article/pii/0042698989901338.

[13] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. Large-scale video classification with convolutional neural networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.

[14] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[15] Bence P Ölveczky, Stephen A Baccus, and Markus Meister. Segregation of object and background motion in the retina. *Nature*, 423(6938):401–408, 2003.

[16] Bence P Ölveczky, Stephen A Baccus, and Markus Meister. Retinal adaptation to object motion. *Neuron*, 56(4):689–700, 2007.

[17] Daniela Pamplona and Alexandre Bernardino. Smooth foveal vision with gaussian receptive fields. In *Humanoid Robots, 2009. Humanoids 2009. 9th IEEE-RAS International Conference on*, pages 223–229. IEEE, 2009.

[18] E. L. Schwartz. Spatial mapping in the primate sensory projection: Analytic structure and relevance to perception. *Biological Cybernetics*, 25(4):181–194, 1977. ISSN 1432-0770. doi: 10.1007/BF01885636. URL http://dx.doi.org/10.1007/BF01885636.

[19] Eric L. Schwartz. Computational anatomy and functional architecture of striate cortex: A spatial mapping approach to perceptual coding. *Vision Research*, 20(8): 645 – 669, 1980. ISSN 0042-6989. doi: http://dx.doi.org/10.1016/0042-6989(80)90090-5. URL http://www.sciencedirect.com/science/article/pii/0042698980900905.

[20] J.P. Siebert, A. Schmidt, G. Aragon-Camarasa, N. Hockings, X. Wang, and W. P. Cockshott. A Biologically Motivated Software Retina for Robotic Vision Applications. In *ECCV 2016 Workshop on Biological and Artificial vision*, October 2016.